

# Single Number - LeetCode

## Java Code:

```
class Solution {
    public int singleNumber(int[] nums) {
        int res = 0;

        for(int i = 0; i < nums.length; i++) {
            res = res ^ nums[i];
        }

        return res;
    }
}
```

## Dry Run & Notes:

Dry Run Example:

Input: nums = [4, 1, 2, 1, 2]

Initial:

res = 0

Step-by-step XOR operation:

res = 0 ^ 4 = 4

res = 4 ^ 1 = 5

res = 5 ^ 2 = 7

res = 7 ^ 1 = 6

res = 6 ^ 2 = 4

Final result: 4 (which is the unique number)

Notes:

- The XOR operator (^) has a key property:
  - $a \oplus a = 0$
  - $a \oplus 0 = a$
- All duplicate numbers cancel each other (e.g.,  $1 \oplus 1 = 0$ ,  $2 \oplus 2 = 0$ )
- The number that appears only once remains in the result after all XOR operations.
- Time Complexity:  $O(n)$
- Space Complexity:  $O(1)$
- Efficient and elegant bit manipulation solution.